

REMARKS

Status of Claims

Applicant respectfully requests reconsideration and allowance of all of the claims of the application. The status of the claims is as follows:

- Claims 1-19, 23-27, 29-38, and 40-46 were pending at the time of the Office Action.
- Claims 12-19, 23-27, 29-38 and 40-46 are canceled herein.
- Claim 1, 5, 6 and 9 are amended herein.
- New claims 47-78 are added herein, directed to the elected invention.
- Following entry of the amendments herein, claims 1-11 and 47-78 will be pending.
- Claims 1, 48, 59 and 70 are independent.

Support for the new claims and the amendments to the claims may be found, for example, at least at paragraphs 0052 and 0084-0093 of Applicant's published application, US2005/0210151. The amendments submitted herein do not introduce any new matter.

Cited Documents

The following documents have been applied to reject one or more claims of the Application:

- **Whiting:** Whiting et al., U.S. Patent No. 5,146,221
- **Vidal:** Vidal et al., U.S. Patent Application Publication No. 2002/0078241

Claims 1-11 and New Claim 47 are Non-Obvious over Whiting in view of Vidal

Claims 1-11 stand rejected under 35 U.S.C. § 103(a) as allegedly being obvious over Whiting in view of Vidal. Applicant respectfully traverses the rejection. Without agreeing that the rejection is proper, and for the purpose of expediting allowance, Applicant has amended claims 1, 5, 6 and 9.

Independent Claim 1

Claim 1, as amended herein, recites, in part:

further compressing the compressed data by encoding the at least one representation that includes the length and the location of the matching sequence, the at least one representation being encoded using a first Huffman table for encoding the length using Huffman encoding;

...

when the location of the matching sequence is not in the LRU table, encoding the location of the matching sequence with Huffman encoding using a second Huffman table, different from the first Huffman table;

using the second Huffman table to also compress literal sequences that have no matching index in the history buffer....

Whiting describes “converting an input data character stream into a variable length encoded data stream in a data compression system” (Whiting, Abstract). “The data compression system includes a history array.” *Id.* “The history array has a plurality of entries and each entry of the history array is for storing a portion of the input data stream.” *Id.* Whiting further describes that “a determination is made as to whether the current string of length MATCHLEN+1 at location HISTORY (NEXT) is equal to the contents of the internal match buffer” (col. 15, lines 11-14). “The internal match buffer contains all MATCHLEN bytes of the currently matching string” (col. 15, lines 14-16).

“This buffer allows faster searching for a new string if the initial attempt to match this string fails” (col. 15, lines 16-17). “An efficiency occurs because the bytes to be matched are immediately available within the chip instead of having to reach them from RAM each time a match is performed” (col. 15, lines 17-21). “Stated differently, the matching buffer acts as a look aside buffer to efficiently enhance processing” (col. 15, lines 21-22). “The match buffer is of finite length (MAXSTR=8 bytes in the preferred embodiment)” (col. 15, lines 22-24).

Vidal describes a “method for transferring an at least partly compressible file from a server computer to a user computer” (Vidal, Abstract). Vidal also describes that the compress program may “determine whether a file is compressible, and if so, which compression algorithm should optimally be used” (Vidal, par. 0042). Vidal states that “the compress program may read a block of data [and try] to compress the block using various algorithms, for example, either or both of the above described dictionary and Huffman encoding algorithms.” *Id.* Vidal further describes that the “compress program may also try various combinations of the algorithms described above and/or other algorithms.” *Id.* “Based on these trials, the compress program selects the algorithm or combination of algorithms which yields the highest compression ratio.” *Id.*

However, the combination of Whiting and Vidal does not teach or suggest “further compressing the compressed data by encoding the at least one representation using a first Huffman table for encoding the length using Huffman encoding...[and]...encoding the location of the matching sequence with Huffman encoding using a second Huffman table, different from the first Huffman table,” as recited in Applicant’s claim 1. Instead, Vidal merely describes that the compress

program may read a block of data and try to compress the data using various algorithms, “for example, either or both of the above described dictionary and Huffman encoding algorithms” (Vidal, par. 0042). Vidal further describes that the “compress program may also try various combinations of the algorithms described above and/or other algorithms.” *Id.* “Based on these trials, the compress program selects the algorithm or combination of algorithms which yields the highest compression ratio.” *Id.* However, Vidal does not teach or suggest further compressing compressed data **using a first Huffman table** and “when the location of the matching sequence is not in the LRU table, encoding the location of the matching sequence with Huffman encoding **using a second Huffman table, different from the first Huffman table**” (emphasis added). Whiting fails to compensate for the deficiencies in Vidal pointed out above.

Further, the combination of Whiting and Vidal does not teach or suggest “using the second Huffman table to also compress literal sequences that have no matching index in the history buffer,” as also presently recited in Applicant’s amended claim 1. For example, Vidal merely describes that the “compress program may also try various combinations of the algorithms described above and/or other algorithms.” *Id.* Applicant respectfully notes that neither this portion of Vidal, nor the remainder of Vidal, teaches or suggests this recitation of Applicant’s amended claim 1. Whiting also fails to make up for these shortcomings in Vidal. Consequently, the combination of Whiting and Vidal does not teach or suggest at least the above recitations of Applicant’s claim 1.

In the rejection of claim 1, the Office concedes that Whiting does not explicitly teach further compressing compressed data (Office Action, p. 5). Instead, the Office cites Vidal, paragraphs 0041 and 0042, respectively, as teaching, “further compressing

the compressed data by encoding the at least one representation that includes the length and the location of the matching sequence, the at least one representation being encoded using a first Huffman table for encoding the length using Huffman encoding” and “when the location of the matching sequence is not in the LRU table, encoding the location of the matching sequence with Huffman encoding using a second Huffman table, different from the first Huffman table.” However, Applicant respectfully notes that paragraphs 0041 and 0042 of Vidal merely describe the following:

[0041] In a preferred embodiment of the invention, the compression algorithm may operate as follows. First, the algorithm may search for symbols (characters) that appear more frequently than others in the data. Then, the algorithm may build a tree, for example a Huffman Tree, as is known in the art, that stores the symbols and their respective frequencies. Each symbol may then be assigned with a code whose length is inversely proportional to the frequency of the symbol. For example if the symbol `A` appears significantly more frequently than the symbol `B`, then the code length of `A` should be shorter than that of `B`. During the encoding process, the algorithm may write the symbol code into memory, instead of the data corresponding to the symbol. This code includes information sufficient to reconstruct the symbol and its frequency. Thus, during decoding, the code may be read and the Huffman tree may be used to reconstruct the original data, as is known in the art. The tree itself may be transferred together with the compressed data, as part of the general compression header as is known in the art.

[0042] In order to determine whether a file is compressible, and if so, which compression algorithm should optimally be used, the compress program may perform the following. First, the compress program may read a block of data having a predetermined size, for example, a block size of about 7 Kilobytes has been found suitable for 56 Kbps modem

communication links. Then, the compress programs tries to compress the block using various algorithms, for example, either or both of the above described dictionary and Huffman encoding algorithms. The compress program may also try various combinations of the algorithms described above and/or other algorithms. Based on these trials, the compress program selects the algorithm or combination of algorithms which yields the highest compression ratio. At this point, the compress program may determine whether the compression ratio is sufficiently high to warrant compression of the entire file, as indicated at block 60, by determining if the compression ratio exceeds a preset threshold. The threshold may correspond to a reduction of a predetermined percentage of the amount of data in the chunk due to compression, for example, a 5% reduction, or any other threshold that yields optimal results based on experimentation with a specific file format. If the compression ratio is determined to be sufficiently high, the compress program proceeds to write the compressed data block onto the disk. However, if the compression ratio is not sufficient, i.e., the size of the original file may not be significantly reduced, then the compress program may write the original data block to disk.

From a review of the foregoing portions of Vidal, and the remainder of Vidal, Applicant submits that the Office has failed to cite any portion of Vidal that teaches or suggests “further compressing the compressed data by encoding the at least one representation that includes the length and the location of the matching sequence, the at least one representation being encoded using a first Huffman table for encoding the length using Huffman encoding” and “when the location of the matching sequence is not in the LRU table, encoding the location of the matching sequence with Huffman encoding using a second Huffman table, different from the first Huffman table.” Instead, Vidal merely provides a general statement that “various combinations” of algorithms

may be tried (Vidal, par. 0042). Consequently, Vidal provides no enablement regarding “further compressing the compressed data...using a first Huffman table for encoding the length” and “when the location of the matching sequence is not in the LRU table, encoding the location of the matching sequence with Huffman encoding using a second Huffman table, different from the first Huffman table.” The Office has failed to cite any prior art that teaches or suggests the use of multiple Huffman tables for further compressing different portions of compressed data. Accordingly, claim 1 is allowable at least for the elements of the above-discussed recitation.

Additionally, the combination of Whiting and Vidal also fails to teach or suggest, “using the second Huffman table to also compress literal sequences that have no matching index in the history buffer,” as also presently recited in Applicant’s amended claim 1. In rejecting similar language previously in Applicant’s dependent claim 6, the Office cites paragraph 0047 of Vidal as teaching this recitation. However, Applicant respectfully notes that paragraph 0047 of Vidal merely describes the following:

[0047] It should be noted that decompress program 24 is preferably adapted to decompress the data received and provide it to the web browser on the fly, i.e., during streaming. In practice, the decompression component receives a first packet or chunk of compressed data, decompresses it, and sends it to the web browser which can begin displaying the data, as described above. The program may then continue processing the next packet or chunk of data. As the data continues to flow in chunks, each chunk of data is individually decompressed (if necessary) and delivered to the web browser.

From a review of the foregoing portion of Vidal, and the remainder of Vidal, it is apparent that Vidal merely describes a decompress program “adapted to decompress data received and provide it to a web browser on the fly” (Vidal, par. 0047). Consequently, Applicant respectfully submits that Vidal does not teach or suggest “using the second Huffman table to also compress literal sequences that have no matching index in the history buffer.” In fact, the cited portion of Vidal appears to bear little relevance to compressing literal sequences, let alone using the second Huffman table to compress the literal sequences. Whiting does not make up for the shortcomings in the teachings of Vidal pointed out above. Accordingly, Applicant respectfully submits that claim 1 is allowable over the combination of Whiting and Vidal for the elements of this recitation as well.

For at least the reasons presented herein, the combination of Whiting and Vidal does not teach or suggest all of the features of claim 1. Accordingly, Applicant respectfully requests that the Office withdraw the 103 rejection of claim 1.

Dependent Claims 2-11 and New Claim 47

Claims 2-11 and new claim 47 ultimately depend from independent claim 1. As discussed above, claim 1 is allowable over the cited documents. Therefore, claims 2-11 and 47 are also allowable over the cited documents of record at least due to the dependency of these claims from an allowable base claim. These claims may also be allowable for the additional features that each recites.

For example, dependent claim 6, as amended, includes “wherein the Huffman encoding for compressing the literal sequences uses a frequency of occurrences table generated from the data and a variable length string assigned as a prefix to each literal

sequence for uniquely representing the literal sequence.” Vidal, on the other hand, merely discusses a decompress program that is adapted to decompress data received (Vidal, par. 0047). Further, paragraph 0042 of Vidal merely provides a general statement that “various combinations” of algorithms may be tried. Consequently, Applicant respectfully submits that Vidal does not teach or suggest Applicant’s amended claim 6. Accordingly, Applicant respectfully submits that dependent claim 6 is also allowable over the combination of Whiting and Vidal for these reasons.

Additionally, dependent claim 9, as amended, includes “representing the representation as a tuple that includes a backpointer that describes the location of the matching sequence in the history buffer and the length of the matching sequence; employing a cost function to determine if the representation utilizes less memory when stored than the matching sequence, wherein the cost function utilizes a product of a size of the backpointer and the length of the matching sequence; and when the representation utilizes less memory when stored than the matching sequence, forming compressed data that includes the representation.” The Office cites Whiting, col. 5, lines 28-34, as teaching Applicant’s claim 9. However, the cited portion of Whiting merely describes prior compression methods in which before a string is sent, a “compression unit generates the Huffman codes for all the bytes in the string and compares the size of the Huffman codes with the size of the string bits” (Whiting, col. 5, lines 28-31). “Typically the Huffman string escape code is four bits, so it takes 19 bits to represent a string” (Whiting, col. 5, lines 31-33). “The smaller of the two quantities is sent” (Whiting, col. 5, lines 33-34). Thus, Applicant respectfully submits that Whiting does not teach or suggest at least “wherein the cost function utilizes a product of a size

of the backpointer and the length of the matching sequence,” as presently recited in Applicant’s amended claim 9. Vidal fails to compensate for the deficiencies in Whiting pointed out above. Accordingly, Applicant respectfully submits that dependent claim 9 is also allowable over the combination of Whiting and Vidal for these reasons.

Furthermore, new dependent claim 47 includes “when an encoded representation is present in the configured data, the decompression module is configured to decode the representation using the LRU table, the third Huffman table, and the fourth Huffman table, and find the matching sequence in the second said history buffer based on a decoded location and a decoded length indicated by the representation.” Vidal, on the other hand, merely discusses a decompress program that is adapted to decompress data received (Vidal, par. 0047). There is no teaching or suggestion in Vidal of at least “decod[ing] the representation using the LRU table, the third Huffman table, and the fourth Huffman table,” as recited in Applicant’s new claim 47. Accordingly, Applicant respectfully submits that dependent claim 47 is also allowable over the combination of Whiting and Vidal for these reasons.

New Claims 48-78 are Non-Obvious over Whiting in view of Vidal

New claims 48-78 are presented herein directed to the elected invention. Applicant respectfully requests allowance of these claims for the following reasons.

Independent Claims 48, 59 and 70

For at least the reasons discussed above with respect to claim 1, Applicant respectfully submits that the combination of Whiting and Vidal does not teach or suggest all of the features of new independent claims 48, 59 and 70. Accordingly, Applicant respectfully requests that the Office allow these claims.

Dependent Claims 49-58, 60-69 and 71-78.

Claims 49-58, 60-69 and 71-78 ultimately depend from one of independent claims 48, 59 or 70, respectively. As discussed above, claims 48, 59 and 70 are allowable over the cited documents. Therefore, claims 49-58, 60-69 and 71-78 are also allowable over the cited documents of record at least due to the dependency of these claims from an allowable base claim. These claims may also be allowable for the additional features that each recites, as discussed above, for example, with respect to claim 6, 9 and 47. Accordingly, Applicant respectfully requests that the Office allow claims 49-58, 60-69 and 71-78.

Conclusion

For at least the foregoing reasons, all pending claims are in condition for allowance. Applicant respectfully requests reconsideration and prompt issuance of the application. If any issues remain that would prevent allowance of this application, Applicant requests that the Examiner contact the undersigned representative before issuing a subsequent Action.

Respectfully Submitted,

Lee & Hayes, PLLC
Representative for Applicant

/Colin D. Barnitz 35061/

Dated: December 15, 2010

Colin D. Barnitz
(colin@leehayes.com; 512-505-8162 x5002)
Registration No. 35061

Damon J. Kruger
(damonk@leehayes.com; 206-876-6018)
Registration No. 60400